

Guía de Ejecución de la Aplicación: Documentación e instrucciones

[Introducción](#)

[Configuración del entorno](#)

[PostgreSQL](#)

[RabbitMQ](#)

[Docker \(opcional\)](#)

[Instalación de FFmpeg](#)

[Variables de entorno](#)

[Desarrollo](#)

[Producción](#)

[Instancias de VM de Google Cloud](#)

[Ejecutar la aplicación de forma local](#)

[Modo desarrollo](#)

[Modo producción](#)

[Levantar el entorno completo con docker](#)

[Realizar peticiones a entorno desplegado](#)

Introducción

A continuación, se presenta una lista de pasos para ejecutar el entorno completo de la aplicación de edición de videos. Se expondrán diferentes maneras de interactuar tanto con los entornos virtuales como de forma local, instalando las herramientas necesarias para el funcionamiento de la aplicación.

Para poder esta versión de la entrega, es necesario desplazarse a la siguiente rama de git

```
git checkout release_entrega_2
```

Configuración del entorno

PostgreSQL

Asegúrate de tener PostgreSQL instalado y configurado en tu sistema. Puedes descargarlo desde [el sitio oficial de PostgreSQL](#).

RabbitMQ

Instala RabbitMQ siguiendo las instrucciones de [la documentación oficial de RabbitMQ](#).

O en la ruta raíz del proyecto se encuentra un archivo llamado `rabbitmq-compose.yaml` para montar de manera local RabbitMQ.

Docker (opcional)

Puedes descargar Docker desde [el sitio oficial de Docker](#).

Instalación de FFmpeg

FFmpeg es una herramienta de línea de comandos para grabar, convertir y reproducir audio y video. Es una herramienta poderosa y versátil para trabajar con multimedia en sistemas operativos Linux, Windows y macOS.

- **Linux:**

```
sudo apt-get install ffmpeg
```

- **macOs:**

```
brew install ffmpeg
```

- **Windows:**

1. Descarga un archivo zip de FFmpeg desde el [sitio oficial de FFmpeg](#).
2. Extrae el contenido del archivo zip en una carpeta de tu elección.

3. Agrega la ruta de la carpeta de FFmpeg al PATH del sistema para poder ejecutar FFmpeg desde cualquier ubicación en la línea de comandos.

Variables de entorno

Desarrollo

#DB

SQL_DOMAIN=localhost
SQL_USER=postgres
SQL_PWD=postgres
SQL_DB=drl_cloud

#RABBITMQ

RABBITMQ_API_HOST=localhost
RABBITMQ_WORKER_HOST=localhost
RABBITMQ_QUEUE_NAME=video_processing_tasks
RABBIT_ADMIN_USER=guest
RABBIT_ADMIN_PASSWORD=guest

#LOCAL FILES

LOGO_VIDEO_ITEM_NAME=LOGO_VIDEO.mp4
LOGO_NAME=CIRCULAR_IDRL_LOGO.png
VIDEO_FOLDER_NAME=videos
ASSETS_PATH=assets
LOGO_FOLDER_NAME=logo
OUTPUT_VIDEO_NAME=OUTPUT.mp4
GLOBAL_VIDEO_SIZE=1280:720
JWT_SECRET_KEY=development

```
#SERVER FTP
```

```
FTP_REMOTE_SERVER=35.226.186.200
```

```
FTP_ADMIN_USER=ftpadmin
```

```
FTP_PASSWORD=misoftp
```

```
FTP_VIDEOS_FOLDER=videos
```

```
# API
```

```
API_HOST=0.0.0.0
```

```
API_PORT=5000
```

Producción

```
#DB
```

```
SQL_DOMAIN=10.69.80.3
```

```
SQL_USER=postgres
```

```
SQL_PWD=misw4204
```

```
SQL_DB=postgres
```

```
#RABBITMQ
```

```
RABBITMQ_API_HOST=10.0.0.6
```

```
RABBITMQ_WORKER_HOST=localhost
```

```
RABBITMQ_QUEUE_NAME=video_processing_tasks
```

```
RABBIT_ADMIN_USER=misoworker
```

```
RABBIT_ADMIN_PASSWORD=misoworker
```

```
#LOCAL FILES
```

```
LOGO_VIDEO_ITEM_NAME=LOGO_VIDEO.mp4
```

```
LOGO_NAME=CIRCULAR_IDRL_LOGO.png
```

```
VIDEO_FOLDER_NAME=videos
```

```
ASSETS_PATH=assets
```

```

LOGO_FOLDER_NAME=logo
OUTPUT_VIDEO_NAME=OUTPUT.mp4
GLOBAL_VIDEO_SIZE=1280:720
JWT_SECRET_KEY=production

#SERVER FTP

FTP_REMOTE_SERVER=10.0.0.5
FTP_ADMIN_USER=ftpadmin
FTP_PASSWORD=misoftp
FTP_VIDEOS_FOLDER=videos

# API

API_HOST=0.0.0.0
API_PORT=5000

```

En este caso, los host de las herramientas poseen una ip interna, ya que este es el archivo de configuración que se usa para el despliegue en las maquinas virtuales, si se desea ejecutar de forma local pero con comunicación a las instancias virtuales, se deberan cambiar los valores con ip interna por ip externa, en la tabla se hace una pequeña aclaración de donde vienen las ip y de como cambiarlas:

Instancias de VM de Google Cloud

Nombre	IP Interna	IP Externa	Variable de entorno
Servidor FTP	10.0.0.5	35.226.186.200	FTP_REMOTE_SERVER
Worker y RabbitMq	10.0.0.6	34.68.134.247	RABBITMQ_API_HOST
PostgreSQL	10.69.80.3	34.27.239.157	SQL_DOMAIN

En la siguiente ruta se podrá ingresar a la interfaz gráfica de RabbitMQ montada en el servidor del Worker, se deberá usar el siguiente usuario y contraseña:

```

Usuario=misoworker
Contraseña=misoworker

```

```
Ruta: http://34.68.134.247:15672
```

```
Queue: video_processing_tasks
```

Ejecutar la aplicación de forma local

Es necesario mencionar que dependiendo de como se ejecute el entorno, la aplicación tomará las variables de entorno pertinentes para su ejecución, en caso de ejecutarse en modo de desarrollo, se tomará el archivo de:

```
proyecto_misw4204/app/dev.env
```

Y en caso de ejecutarse en producción tomará el siguiente archivo:

```
proyecto_misw4204/app/prod.env
```

Modo desarrollo

Para ejecutar la aplicación en modo desarrollo, sigue estos pasos:

1. En una terminal, navega hasta la carpeta `app` del proyecto.
2. Ejecuta el siguiente comando para iniciar la API Flask en modo desarrollo:

```
python app.py dev
```

3. En otra terminal, navega hasta la carpeta `app` del proyecto.
4. Ejecuta el siguiente comando para iniciar el worker en modo desarrollo:

```
python worker.py dev
```

Modo producción

Para ejecutar la aplicación en modo producción, sigue estos pasos:

1. En una terminal, navega hasta la carpeta `app` del proyecto.
2. Ejecuta el siguiente comando para iniciar la API Flask en modo producción:

```
python app.py
```

3. En otra terminal, navega hasta la carpeta `app` del proyecto.

1. Ejecuta el siguiente comando para iniciar el worker en modo producción:

```
python worker.py
```

Levantar el entorno completo con docker

En la carpeta raíz se encuentra un archivo Docker compose, el cual se llama **docker-compose.yaml**. Para levantar la aplicación solo es necesario ejecutar el siguiente comando:

```
docker-compose up --build
```

Realizar peticiones a entorno desplegado

A la siguiente ip externa se se podrán hacer todas las peticiones, actualmente la Api esta desplegada en esta VM y esta en funcionamiento:

```
34.132.92.148:5000
```

Para más información de los endpoints de la Api verificar los siguientes documentos:

- [Autenticación](#)
- [Gestión de Tareas](#)